

The Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (previously presented) An interface for transactions among nodes in a network including a plurality of nodes which execute processes involved in the transactions, the interface being stored in a computer readable medium, comprising:
a machine readable specification of an interface to transaction processes stored in memory accessible by at least one node in the network, including interpretation information providing a definition of an input document, and a definition of an output document, the definitions of the input and output documents comprising respective descriptions of sets of storage units and logical structures for the sets of storage units.

2. (original) The interface of claim 1, wherein the interpretation information includes data type specifications for at least one logical structure in the definitions of the input and output documents.

3. (original) The interface of claim 1, wherein the interpretation information includes at least one data structure mapping predefined sets of storage units for a particular logical structure in the definitions of the input and output documents, to respective entries in a list.

4. (original) The interface of claim 1, including a repository in memory accessible by at least one node in the network storing a library of logical structures, and interpretation information for logic structures.

5. (original) The interface of claim 1, wherein the machine readable specification includes a document compliant with a definition of an interface document including logical structures for storing an identifier of a particular transaction, and at least one of definitions and references to definitions of input and output documents for the particular transaction.

6. (original) The interface of claim 1, wherein the machine readable specification includes a document compliant with a definition of an interface document including logical structures for storing an identifier of the interface, and for storing at least one of specifications and references to specifications of a set of one or more transactions supported by the interface.

7. (original) The interface of claim 6, wherein the machine readable specification includes a reference to a specification of a particular transaction, and the specification of the particular transaction includes a document including logical structures for storing at least one of definitions and references to definitions of input and output documents for the particular transaction.

8. (original) The interface of claim 1, wherein the storage units comprise parsed data.

9. (original) The interface of claim 8, wherein the parsed data in at least one of the input and output documents comprises:

character data encoding text characters in the one of the input and output documents, and

markup data identifying sets of storage units according to the logical structure of the one of the input and output documents.

10. (original) The interface of claim 9, wherein at least one of the sets of storage units encodes a plurality of text characters providing a natural language word.

11. (original) The interface of claim 8, wherein the interpretation information for at least one of the sets of storage units identified by a particular logical structure of at least one of the input and output documents, encodes respective definitions for sets of parsed characters.

12. (original) The interface of claim 8, wherein the storage units comprise unparsed data.

13. (original) The interface of claim 1, including a repository stored in memory accessible by at least one node in the network of document types for use in a plurality of transactions, and wherein the definition of one of the input and output documents includes a reference to a document type in the repository.

14. (original) The method of claim 13, wherein the repository of document types includes a document type for identifying participant processes in the network.

15. (original) The interface of claim 1, wherein the definitions of the input and output documents comprise document type definitions compliant with a standard Extensible Markup Language XML.

16. (original) The interface of claim 1, wherein the machine readable data structure including interpretation information comprises a document organized according to a document type definition compliant with a standard Extensible Markup Language XML.

17. – 60. (cancelled).

61. (original) A method for programming a commercial transaction in a network, comprising:

defining a machine readable definition of an input document for a node in the network including resources to execute a process in the transaction, and a machine readable definition of an output document for the node, the definitions of the input and output documents comprising respective descriptions of sets of storage units and logical structures for the sets of storage units; and

providing interpretation information for the logical structures to the node.

62. (original) The method of claim 61, wherein the interpretation information includes data type specifications for at least one logical structure in the definitions of the input and output documents.

63. (original) The method of claim 61, wherein the interpretation information includes at least one data structure mapping predefined sets of storage units for a particular logical structure in the definitions of the input and output documents, to respective entries in a list.

64. (original) The method of claim 61, the step of providing interpretation information includes providing a repository in memory accessible by at least one node in the network storing a library of logical structures, and interpretation information for logic structures.

65. (original) The method of claim 61, including defining a machine readable specification of an interface including a document compliant with a definition of an interface document including logical structures for storing an identifier of a particular

transaction, and at least one of the definitions and references to the definitions of the input and output document.

66. (original) The method of claim 61, wherein the storage units comprise parsed data.

67. (original) The method of claim 66, wherein the parsed data in at least one of the input and output documents comprises:

character data encoding text characters in the one of the input and output documents, and

markup data identifying sets of storage units according to the logical structure of the one of the input and output documents.

68. (original) The method of claim 67, wherein at least one of the sets of storage units encodes a plurality of text characters providing a natural language word.

69. (original) The method of claim 67, wherein the interpretation information for at least one of the sets of storage units identified by a particular logical structure of at least one of the input and output documents, encodes respective definitions for sets of parsed characters.

70. (original) The method of claim 66, wherein the storage units comprise unparsed data.

71. (original) The method of claim 61, wherein the definitions of the input and output documents comprise document type definitions compliant with a standard Extensible Markup Language XML.

72. (original) The method of claim 61, including:
providing a parser to generate event signals in response to logical structures in the definition of the input document; and
providing event listener programs which respond to the event signals to execute the process.